NRL Report 8463

Image Sampling and Interpolation

W. H. CARTER

Applied Optics Branch
Optical Sciences Division

April 1, 1981

NAVAL RESEARCH LABORATORY
Washington, D.C.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Report 8463	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* IMAGE SAMPLING AND INTERPOLATION		5. TYPE OF REPORT & PERIOD COVERED 1 May-31 July 1980
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) W. H. Carter		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, DC 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS RR 011 09 01; 65-1156-0-1
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research 800 N. Quincy Street Arlington, VA 22217		12. REPORT DATE April 1, 1981
		13. NUMBER OF PAGES 25
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		15. SECURITY CLASS. *(of this report)* UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Image processing
Splines
Interpolation

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

Various methods for image sampling and interpolation are considered. The use of Nyquist sampling is discussed; interpolation of sampled data by the use of *pp*-functions and *B*-splines is introduced; and the various methods are compared for the case of image sampling. An elementary introduction to *B*-splines and their applications is given in a manner that lacks rigor but which should appeal to engineers.

CONTENTS

IMAGE SAMPLING AND INTERPOLATION

I. INTRODUCTION

This report records the results of a 3-month study during the summer of 1980 made by the author while on temporary assignment to the Office of Naval Research. The study was begun at the suggestion of Dr. William J. Condell, Jr., Head of the Physics Program at ONR.

This study includes a review of spline functions, in particular B-splines, and a consideration of their use for the foveal sampling of images. Such a study is of interest to the Navy. In many applications of image sampling, as for example in a missile warhead, only a very few samples can be handled. Thus methods of taking these samples in a more efficient manner are necessary. It was thought that B-splines might be useful because they may be used for interpolating data not sampled at uniformly spaced points. This study confirms this idea but indicates that even when using splines, uniformly spaced data make numerical interpolation more efficient.

Because of the limited time available and the newness of the material on B-splines to the author, all of the possibilities could not be fully developed. In particular no actual numerical work with images was done. Without actual experience with images no study of image sampling and interpolation can be taken very far. The scope of this report is therefore limited. In it the author reviews the well-known Nyquist sampling theorem and points out its drawbacks in Section II in order to motivate the use of splines. In Section III the basic properties of B-splines are introduced in a very different manner than is usual in the literature. While this material lacks rigor it should be much simpler for the nonmathematician to follow. The use of the so called pp-representation and B-representation are presented in Section IV as extensions of the familiar cubic spline methods. And finally in Section V a very brief account is given as to how all of this might be applied to image sampling and the foveal problem. The conclusions are briefly stated in Section VI.

II. SAMPLING AND INTERPOLATION OF BANDLIMITED IMAGES

An image is defined in this report as a two-dimensional distribution of intensity described by the function $I(x,y)$. This intensity distribution may be a primary image obtained from light scattered directly from a scene say as in the back focal plane of a camera, or it may be a secondary image formed by light using an electronic or photographic recording of the primary image.

If the function $I(x,y)$ has finite support such that

$$I(x,y) = 0 \text{ if } |x| > a \text{ or } |y| > b\ , \tag{2.1}$$

Manuscript submitted November 19, 1980.

then the image is bounded. All images formed by practical instruments are bounded. If, on the other hand, the function $I(x,y)$ has a Fourier transform

$$\tilde{I}(\xi,\eta) = \iint_{-\infty}^{\infty} I(x,y)\, e^{-2\pi i(\xi x+\eta y)} dx dy\ , \tag{2.2}$$

which has finite support, i.e.,

$$\tilde{I}(\xi,\eta) = 0 \text{ if } |\xi| > \alpha \text{ or } |\eta| > \beta\ , \tag{2.3}$$

then the image is bandlimited. By the Paley-Wiener theorem of Fourier analysis an image cannot be both bounded and bandlimited [1]. However, this theorem reflects a basic difficulty in the simple mathematical model (or in achieving ideal optical instruments — depending on your point of view) because images do appear to be both bounded and bandlimited.

Images are bounded because all imaging systems contain apertures which block light outside of some finite area in the image plane. Images are bandlimited because all imaging systems have a finite resolving power and can in no case resolve image detail smaller than about a wavelength of light. In the following we assume that an image is either bandlimited or that it is bounded but not both so that we may use this mathematical model.

We will treat the case of a linear stationary imaging system. Thus a bounded image can be represented by

$$\begin{aligned} I(x,y) &= \iint_{-\infty}^{\infty} h(x'-x,y'-y)\, 0(x',y')\, dx'dy' \text{ if } |x| \leqslant a \text{ and } |y| \leqslant b\ , \\ &= 0 \text{ otherwise,} \end{aligned} \tag{2.4}$$

where $h(x,y)$ is the spread function (the image of a point), and $0(x,y)$ is an unbounded and unbandlimited "perfect" image. In Eq. (2.4) $I(x,y)$ is bounded but not bandlimited. We can represent a bandlimited image by

$$I'(x,y) = \iint_{-\infty}^{\infty} h(x'-x,y'-y)\, 0'(x',y') dx'dy' \text{for all } x \text{ and } y\ , \tag{2.5}$$

where

$$\begin{aligned} 0'(x',y') &= 0(x',y') \text{ if } |x| \leqslant a \text{ and } |y| \leqslant b\ , \\ &= 0 \text{ otherwise.} \end{aligned}$$

Now $I'(x,y)$ represents an image that is bandlimited but not bounded. In the rest of this section we will consider the bandlimited image described by $I'(x,y)$.

To record an image it is often convenient (or necessary) to sample it. For example, a vidicon samples an image continuously along horizontal raster lines so that a two-dimensional spatial function $I(x,y)$ can be represented by a one-dimensional temporal function $v(t)$, the video signal. As a second example, a photographic emulsion contains silver halide crystals which collect light and are converted by development into free silver. Thus the crystals sample the image intensity $I(x,y)$ in a complicated manner. An image can be sampled in many other ways that can be selected at will by the instrument designer. A convenient choice for many applications [2] is to sample the image over a rectangular mesh, i.e.,

$$\overline{I}(x,y) = \sum_{\substack{n=-N \\ m=-M}}^{N,M} I'(x,y)\delta(x - nd_x)\delta(y - md_y) \tag{2.6}$$

so that the sampled image is represented by the discrete data values I_{nm} where

$$\begin{aligned} \overline{I}(x,y) &= I_{nm} \quad \text{if } x = nd_x,\ y = md_y\ , \\ &= 0 \text{ otherwise,} \end{aligned} \tag{2.7}$$

where d_x, d_y are the constant sampling intervals, and where δ represents the Dirac delta function. Data sampled in this way are often referred to as gridded data in the interpolation literature, and we will use this expression in this report.

To record or transmit the sampled image we need only deal with the sequence of real numbers I_{nm} rather than the continuous function $I'(x,y)$. This data reduction is essential for many operations on the image as, for example, digital processing by a computer.

It is critically important to the accurate and efficient representation of the image that the sampling intervals have the values

$$d_x = \frac{1}{2\alpha} \text{ and } d_y = \frac{1}{2\beta}\,. \tag{2.8}$$

These equations are equivalent to the Nyquist criterion which states that a bandlimited signal must be sampled at uniformly spaced points separated by half the period of the highest Fourier component in the spectrum of the signal. It is shown in Appendix A that if the values of d_x and d_y are larger than specified by Eq. (2.8), then aliasing errors occur which prevent accurate recovery of the image from the sampled data. If, on the other hand, d_x and d_y are made smaller than specified by Eq. (2.8), then the extra samples obtained are wasted because no additional information about the image is obtained.

If the image is bandlimited and properly sampled, then as shown in Appendix A the image can be recovered exactly by using the formula

$$I'(x,y) \underset{\substack{N\to\infty \\ M\to\infty}}{\sim} \sum_{\substack{n=-N \\ m=-M}}^{N,M} I_{nm} \frac{\sin\left[\frac{\pi}{d_x}\left(x - nd_x\right)\right]}{(x - nd_x)} \; \frac{\sin\left[\frac{\pi}{d_y}\left(y - md_y\right)\right]}{(y - md_y)}\,. \tag{2.9}$$

We will call this sampling and interpolation procedure Nyquist sampling. Nyquist sampling allows the image $I'(x,y)$ to be recovered at every point from just the sampled values I_{nm}. Clearly some arbitrary function $f(x,y)$ contains much more information than any discrete, finite set of sampled values f_{nm}. Therefore we must conclude from Eq. (2.9) that the condition that $I'(x,y)$ be bandlimited is a very strong condition which greatly limits the information content of this function.

Often it is not possible to properly sample an image. Usually the number of samples that can be taken, stored, and perhaps transmitted in some fashion is very limited. So if the image is not properly bandlimited before sampling, aliasing problems develop. This well-known problem is discussed in Appendix A and has been examined experimentally [3]. It appears that a badly aliased image is usually not useful. Therefore a completely different approach to image sampling and interpolation is needed if the original image cannot be properly bandlimited. Thus the requirement that an image be properly bandlimited is a serious limitation for Nyquist sampling.

A second limitation to this sampling and interpolation method is that it is restricted to gridded data. It is often more useful to employ a limited number of possible samples in a more efficient manner as is done by the human eye. In the eye samples are taken much more closely together over a small region near the center of the field-of-view called the fovea than they are elsewhere. The result is that we have a much better ability to resolve fine detail in the foveal image than elsewhere in our field of view. Samples are taken so far apart near the edges of the field of view that we can detect little more than motion there. In this manner our eyes make the best use of a limited number of photoreceptors.

In this report we are particularly concerned with optimizing the use of a limited number of samples. To do this using Nyquist sampling it is necessary to divide the image into square subareas and to take gridded data from each area. The sampling intervals may be allowed to differ from one area to another but must be the same within each area. However there remains a serious problem. It is very difficult to individually bandlimit each subarea of the image so that the Nyquist criterion is satisfied there. Usually the entire image must be bandlimited in the same way. Then if the image is properly bandlimited for the most finely sampled area, it will be aliased elsewhere, or if the image is properly bandlimited for the least finely sampled area, the extra samples elsewhere will be wasted and the interpolated image would be uniformly poor. Thus Nyquist sampling is not usually of much use under these conditions.

If we cannot bandlimit an image properly so that, over every region, the image satisfies the Nyquist criterion locally, then Nyquist sampling should not be used. Fortunately there is a useful alternative. It is always possible to sample an image at any set of points and then to reconstruct an estimate of the image from these data. It is not possible in general to reconstruct the original image precisely as it is by the use of Nyquist sampling; however, it is often possible to construct a useful approximation.

A common method for reconstructing a one-dimensional function $f(x)$ from scattered data (samples taken at random) is to set

$$f(x) = f_n \text{ , where } x = x_n \text{ ,} \tag{2.10}$$

so that at the original sample points x_n the function $f(x)$ has its original values and then to use a draftsman's spline (a tool like a rubber ruler) to continue the data smoothly in between. This gives a much better representation of the original function than does Nyquist sampling with serious aliasing. As a practical matter for a two-dimensional function it is better to employ the numerical methods of fitting surfaces to scattered data. Here we represent the draftsman's tool by a piecewise polynomial "spline" function which is used to connect the data points in a satisfactory manner.

In the following sections we introduce some of the fundamentals of spline interpolation and discuss the application of these methods to image interpolation.

III. *B*-SPLINE FUNCTIONS

B-splines or basis splines were first introduced by Schoenberg [4] as a set of functions defined by

$$M_k(t) := \frac{1}{2\pi}\int_{-\infty}^{\infty}\left(\frac{\sin(\xi/2)}{\xi/2}\right)^k e^{-i\xi t}d\xi \, . \qquad (3.1)$$

For $k = 1$ it is well known to mathematical physicists that

$$\begin{aligned} M_1(t) &= FT\left(\frac{\sin(\xi/2)}{\xi/2}\right) = \text{Rect}\ (t) \\ &= 1 \ \text{ if } |t| \leqslant \frac{1}{2} \qquad (3.2) \\ &= 0 \ \text{ otherwise,} \end{aligned}$$

where FT represents a Fourier transform. It is also well known that by use of the convolution theorem Eq. (3.1) can be rewritten

$$M_k(t) = FT\left[\left(\frac{\sin(\xi/2)}{\xi/2}\right)^{k-i}\right] \circledast FT\left[\left(\frac{\sin(\xi/2)}{\xi/2}\right)^{i}\right], \qquad (3.3)$$

so that $M_k(t)$ can be built up from the Rect (t) function via $k - 1$ convolutions. It is clear from this that because Rect (x) has small support (i.e., vanishes outside of the domain $-1/2 \leqq t \leqq 1/2$) that all of the $M_k(t)$ built up by repeated convolution with Rect (t) will similarly have small support (i.e., vanish outside of the domain $-k/2 \leqq t \leqq k/2$). This is one of the most useful properties of the *B*-splines ([5], p. 109).

The *B*-splines up to cubic order are given in Fig. 3.1 for the case of uniform sampling. Note that $M_k(t)$ is a peacewise polynomial (pp) function of order $k - 1$. It is easily demonstrated that $M_k(t)$ is continuous with continuous derivatives up to order $k - 2$. Higher order derivatives are continuous everywhere but at the break points between polynomial segments. These break points are called knots in the literature on *B*-splines.

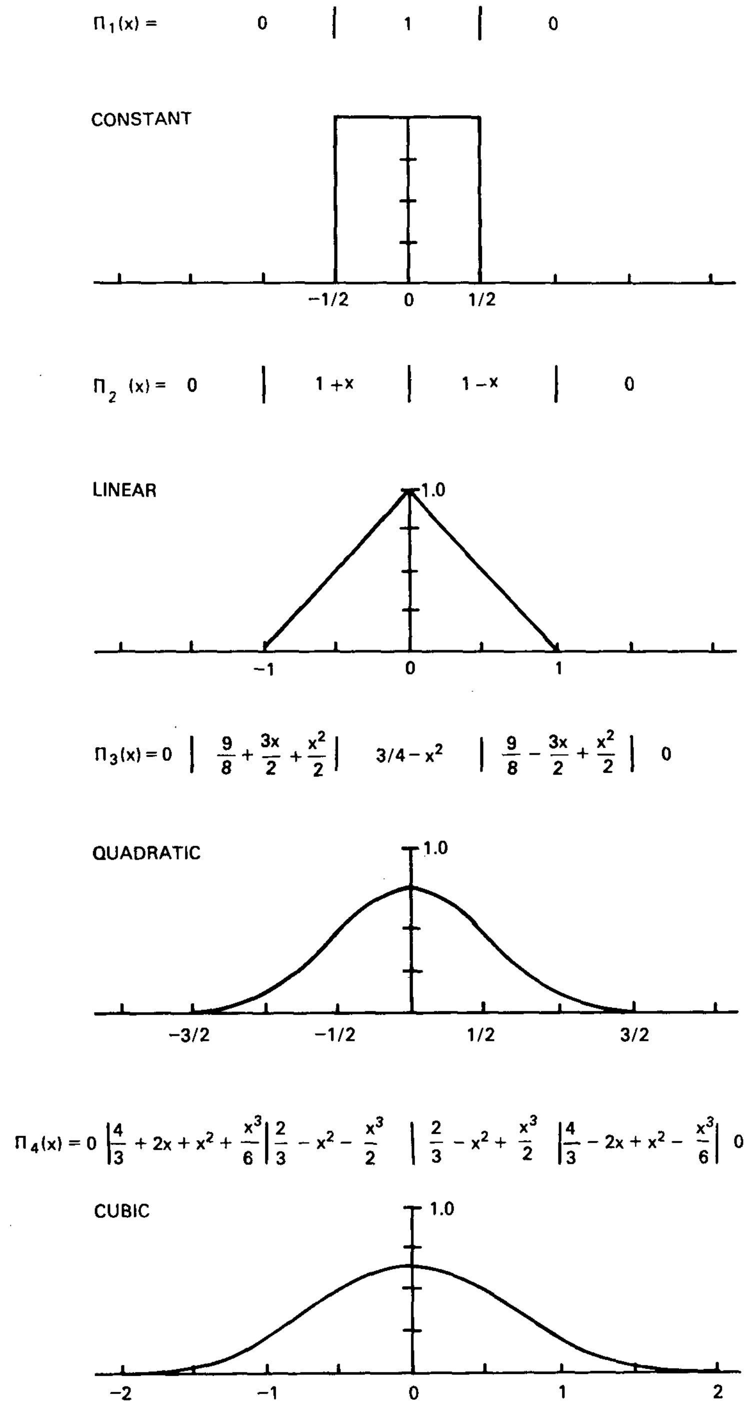

Fig. 3.1 — B-splines $M_k(x)$ on a equispaced knot sequence for $k = 1,2,3,4$

The definition of B-splines given by Eq. (3.1) holds only for the special case of an equispaced knot sequence, that is for polynomial segments of equal length as shown in Fig. 3.1. However, Curry and Schoenberg [6] showed that if the definition of the B-splines is expressed in terms of the kth order divided difference as described here in Appendix B, then the definition can be extended to an arbitrary knot sequence. This admits a much more general class of pp-spline functions composed of polynomial segments of arbitrary length and with discontinuities of arbitrary order at the break points between segments.

To extend B-splines to allow arbitrary knot sequences $[t_1, t_2 \ldots t_\ell]$ we redefine the B-spline of order 1 at location i by

$$\begin{aligned} B_{i,1}(t) &= 1 \text{ if } t_i \leqslant t \leqslant t_{i+1} , \\ &= 0 \text{ otherwise,} \end{aligned} \tag{3.4}$$

which agrees with the definition in Eq. (3.1) and then generate higher order B-splines using the recursion relation (see Appendix B, Eq. (B.17))

$$B_{i,k}(t) = \frac{t - t_i}{t_{i+k-1} - t_i} B_{i,k-1}(t) + \frac{t_{i+k} - t}{t_{i+k} - t_{i+1}} B_{i+1,k-1}(t) . \tag{3.5}$$

This is equivalent to the definition of $B_{i,k}(t)$ in the mathematical literature which depends on the rather complicated concept of the divided difference (see Appendix B). If the knot sequence $[t_1, t_2 \ldots t_\ell]$ is monotonically increasing with constant interval between knots, Eq. (3.5) is equivalent to the definition in Eq. (3.1). Otherwise it is not.

B-splines are very important in the theory of pp-functions because they form a basis for such functions (hence the name basis splines). Any pp-function can then be represented as a superposition of B-splines of the form ([5], pp. 119-120):

$$f_k(t) = \sum_{i=1}^{n} \alpha_i B_{i,k}(t) \tag{3.5}$$

or if $t_j \leqslant t \leqslant t_{j+1}$ by

$$f_k(t) = \sum_{i=j-k+1}^{j} \alpha_i B_{i,k}(t) \tag{3.6}$$

where $f_k(t)$ is a pp spline function of order k for the arbitrary knot sequence $[t_1, t_2 \ldots t_\ell]$, and

$$n = k\ell - \sum_{i=2}^{\ell} \nu_i . \tag{3.7}$$

The knots are a sequence of points between the polynomial segments of the B-splines. They may be coincident as indicated by the parameter $\nu_i(\leqslant k)$ in such a manner that at some point t there are $(k - \nu_i)$ knots t_i. To use a pp-function to interpolate sampled data the knot may represent the location of data samples. They may be separated by arbitrary intervals corresponding to nonuniform sampling intervals. The coincident knots are useful for specifying the number of continuity conditions the pp-function satisfies over the knot point. At a point with a single knot the pp-function and its derivatives up to the order $k - 1$ must be continuous over the point, but at a point with $k - \nu_i$ knots the pp-function will satisfy only ν_i continuity conditions over the point. Multiple knots are also useful for osculatory interpolation in which we specify data concerning derivatives as well as values of the sampled function.

Interpolation and the use of B-splines are discussed in the following sections.

IV. INTERPOLATION IN ONE DIMENSION

The interpolation of data sampled along a single line is introduced in this section. The more complicated two-dimensional interpolation problem which applies to images is discussed in the next section.

The best known and probably the most commonly used spline interpolation method employs data fitting with piecewise polynomial cubic splines. The application of this method in one dimension is clearly described in many standard works on numerical analysis ([7], p. 474-491). We will describe this method first as an introduction to the more general methods.

For the usual pp-cubic spline interpolation a different cubic polynomial, for example

$$f(x) = \mathrm{a}_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i \,, \tag{4.1}$$

is fitted between each point $x_i(1 \leqslant i \leqslant n)$ at which data have been sampled and the next larger point x_{i+1}. The sample points become break points between these polynomial segments. The coefficients in each segment are determined by making restrictions on the pp-cubic spline. The most usual conditions are that the pp-function:

(1) is continuous over the break points ($n - 2$ equations)

(2) gives the sampled data value at each of the break points (n equations)

(3) has continuous first and second derivatives over the break points ($2(n - 2)$ equations).

Thus for the n sampled data values we obtain $4n - 6$ equations for the $4n - 4$ coefficients a_i, b_i, c_i, d_i in Eq. (4.1). Clearly we require two additional conditions in order to specify the coefficients. These can be made arbitrarily, for example, by specifying the derivatives of the pp-function of each of the two end points. The $4n - 4$ equations are easily reduced to n

equations analytically, and solution of the remaining n equations containing the sampled data values reduces to inverting an $n \times n$ tridiagonal matrix using a computer.

There are several variations to *pp*-cubic spline interpolation that can sometimes give a more satisfactory result in some special cases. One, of course, is to choose differently the two arbitrary restrictions described in the last paragraph. Another is to relax the conditions that the second derivative be continuous over the break points and instead require that the first derivative be specified at the break points by sampled data. Clearly there is a lot of flexibility to *pp*-interpolation. This is brought out more clearly in the generalized methods which we will now briefly introduce.

The *pp*-cubic spline interpolation can be easily modified by using polynomials of higher or lower order. Higher order polynomials of order k (cubic is $k = 4$) can be used to give a *pp*-interpolation function which has continuous derivatives to higher order $k - 2$ over the break points but at the cost of more computation. The matrix is still $n \times n$ but is no longer tridiagonal but has k nonzero semidiagonals instead (k-diagonal). Lower order polynomials greatly reduce the computations but give less smooth *pp*-functions.

Once the interpolation calculations have been carried out, the *pp*-function is stored in the computer for use in calculating values of the function $f(x)$ which was represented originally by the sampled data. To store a *pp*-representative for some function $f(x)$ in a computer we require ([5], p. 88)

(1) the integers k and n giving the order and number of the polynomial pieces, respectively,

(2) the strictly increasing sequence $x_1, x_2, x_3, \ldots x_{\ell+1}$ of its breakpoints (which do not in general have to be equally spaced), and

(3) the matrix

$$C_{ji} = \left. \frac{\partial^{j-1}}{\partial x^{j-1}} \right|_{x=x_i} f(x) \text{ for } 1 \leqslant j \leqslant k, \text{ and } 1 \leqslant i \leqslant n$$

of its (right) derivatives at the breakpoints. Then the *pp*-function can be used to represent the sampled function over each domain $x_i \leqq x \leqq x_{i+1}$ by

$$f(x) = \sum_{m=0}^{k-1} C_{m+1,i} \frac{(x - \xi_i)^m}{m!} . \tag{4.2}$$

This is nothing but a generalization from the cubic spline representation to splines of k order where for $k = 4$ we have $a_i = C_{4i}/6$, $b_i = C_{3i}/2$, $c_i = C_{2i}$, $d_i = C_{1i}$ so that Eqs. (4.1) and (4.2) become the same. Clearly the use of higher order splines provides greater flexibility for improving data interpolation but at the cost of greater computation.

Still greater flexibility can be obtained by the use of an entirely different representation based on the B-splines introduced in the last section. In interpolating a sequence of data points, B-splines (for example as shown in Fig. 3.1, if the data points are equally spaced) of some fixed order k are fitted over k successive knots. Each of the knots (except the free knots discussed later) can represent a sampled data point. The B-splines unlike the polynomial segments described above are allowed to overlap in such a manner that a new B-spline begins at each knot location. Thus we have a rather different sort of interpolation method with additional redundancy which leads to greater flexibility in application.

The calculation of a B-representation for a sampled function $f(x)$ reduces to the inversion of a matrix by a computer. The matrix is again k-diagonal; however, it is now of much higher order due to the extra coefficients required to specify the overlapping B-splines. The order of the matrix is now $n \times n$ where

$$n = k\ell - \sum_i \nu_i \,, \tag{4.3}$$

where k is the order of the B-splines, ℓ is the number of actual data samples, and ν_i is the order of continuity over the knot points (i.e., $\nu_i = 0$, no continuity; $\nu = 1$, continuous $f(x)$ only; $\nu_1 = 2$, continuous $f(x)$ and first derivative of $f(x)$; etc.).

Once the B-representation has been calculated from the sampled data it is stored in the computer. Then the B-representation for some originally sampled function $f(x)$ consists of ([5], p. 119):

(1) the integers k and n ($n = k\ell - \sum_i \nu_i$) giving the order of the B-spline segments in the interpolating function and the number of linear parameters required to represent the function,

(2) the vector

$$t_1 \leqslant t_2 \leqslant t_3 \leqslant \ldots \leqslant t_k \leqslant \xi_1 \leqslant \xi_2 \leqslant \ldots \leqslant \xi_{\ell+1} \leqslant t_{n+1} \leqslant \ldots \leqslant t_{n+k} \tag{4.4}$$

containing the knots (possibly partially coincident) in increasing order, and,

(3) the vector α_i, $1 \leqslant i \leqslant n$ of the coefficients of f with respect to the B-spline basis as shown in Eq. (3.6).

The knots ξ_1 to $\xi_{\ell+1}$ can be actual sample points whereas the free knots t_1 to t_k and t_{n+1} to t_{n+k} are arbitrary and can be chosen to optimize the interpolation. The proper choice of knots is a rather complicated subject ([5], Chap. XIII) that we will not attempt to cover at the level of this report. A proper understanding of this subject gives the computer programmer great flexibility in controlling the properties of the interpolating function.

Once the calculations have been carried out and the B-representation data stored in the computer then the value of the sampled function $f(x)$ can be calculated between each pair of knots $x_i \leqslant x \leqslant x_{i+1}$ by use of Eq. (3.6), i.e.,

$$f(x) = \sum_{i=j-k+1}^{j} \alpha_i B_{i,k}(x) \,. \tag{4.5}$$

In this manner we can calculate a value of $f(x)$ for each point within the domain bounded by the data samples. At the sample points we recover the sampled values and in between we get a smoothly continued set of values which are determined by the data and by the knots chosen by the programmer.

If the original sampled function contains noise such that

$$f(x) = g(x) + n(x) \tag{4.6}$$

where $g(x)$ is the signal and $n(x)$ is the noise, then spline interpolation of samples from $f(x)$ can often be chosen such that the resulting estimate of $f(x)$ is smoothed. For this we choose the interpolation knots to be at the sample points, and we employ spline interpolation with an auxiliary condition that the pp-function minimizes a roughness expression.

In cubic spline interpolation, for example, to smooth the sampled data we replace the usual conditions with the condition that for n sample points the pp-function:

(1) is continuous over the break points ($n - 2$ equations),

(2) gives values a_i at the break points (n equations),

(3) gives continuous first derivatives at the break points ($n - 2$ equations),

(4) gives zero second derivatives at the end points (2 equations),

(5) gives values c_i for the second derivatives at the break points ($n - 2$ equations).

This gives the system of $4n - 4$ equations required to determine the coefficient in Eq. (4.1). The $2n$ parameters a_i and c_i are determined by requiring that the roughness expression ([5], p. 238)

$$R_i(p) = p \sum_{i=1}^{n} \left(\frac{y_i - a_i}{\delta y_i} \right)^2 + \frac{4(1-p)}{3} \sum_{i=1}^{n-1} \Delta x_i \left(c_i^2 + c_i c_{i+1} + c_{i+1}^2 \right) \tag{4.7}$$

be minimized for some value of p ($0 \leqslant p \leqslant 1$), where y_i is the sampled value at the x_i point, δy_i is an estimate of the variance of y_i, and $\Delta x_i = x_{i+1} - x_i$. If we set $p = 0$, the pp-function is optimally smooth but inaccurate, but if we set $p = 1$, the pp-function is accurate (i.e., $a_i = y_i$) but not smooth. In practice a value of p is chosen to give the best trade-off in the opinion of the programmer.

In the next section we consider the extension of one-dimensional interpolation to images.

V. IMAGE INTERPOLATION

The extension of the interpolation methods introduced in the last section from one to two dimensions is not trivial except for gridded data. We are particularly interested in interpolating scattered data in order to implement foveal sampling. That is, we would like to interpolate data which were sampled at points in the original image that were much closer together over the foveal region than outside of it as discussed in Section II. Thus we will consider scattered data first.

Consider an image which has been sampled in the following manner. Let a general curvilinear, orthogonal, two-dimensional coordinate system span the image plane which is assumed to be finite but of arbitrary shape. The coordinate curves must describe a pattern of lines which intersect at right angles as shown in Fig. 5.1. The data samples are taken from the image at points $\xi = \xi_i$ along curves of constant $\eta = \eta_i$. We take ξ_i, η_i to be closely spaced in the foveal region and more widely spaced elsewhere. An example of a set of such sampling points is shown in Fig. 5.2. In this example only 80 samples are shown; however, to sample this area with gridded data and at the foveal sampling rate would require 144 samples. We have reduced the number of samples 44% and maintained the foveal resolution, but at a cost. We cannot precisely recover the image.

To recover an approximation to the image by using the sampled data we employ one-dimensional spline interpolation first along the curves of constant $\xi = \xi_i$ and then, using the interpolated values, along all required curves of constant η. In this manner an estimate of $I(\xi, \eta)$ at any point over the image can be recovered. Since the accuracy of the estimate at a point will improve as the point nears a sampling point, the approximation to the image will be more accurate over the foveal region where the sampling points are closer together.

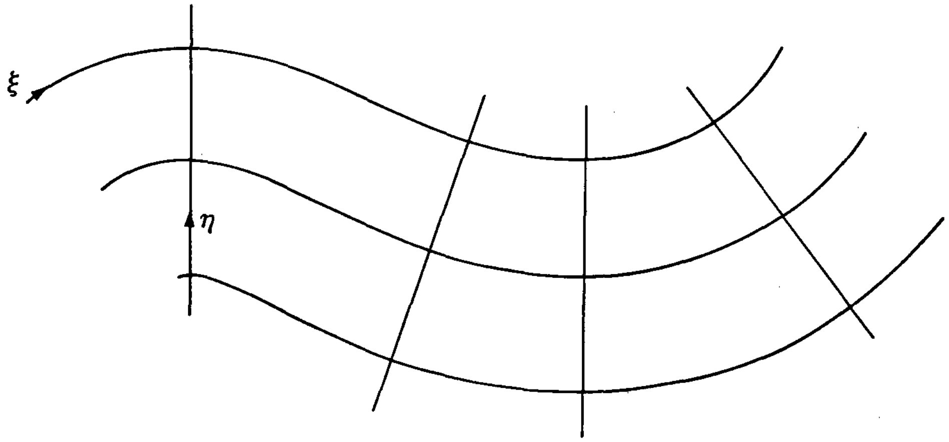

Fig. 5.1 — Coordinate curves on a portion of the image plane

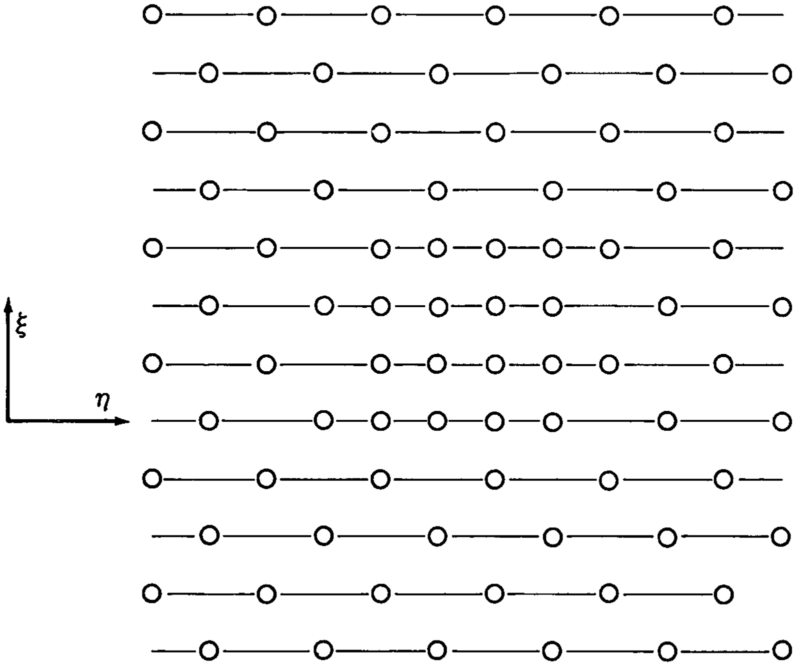

Fig. 5.2 — Example of foveal sampling

Many methods have been invented for interpolation of scattered data in two dimensions. A general review article on this subject was published by Schumaker [8]. The general problem of two-dimensional spline interpolation has not been developed as well as in one dimension. It is always possible to use a two-dimensional *pp*-function over each polygon-shaped region bounded by nearest sample points. In general, the calculation of coefficients representing the *pp*-function cannot be computed very efficiently. The methods used depend on the geometry of the sampling points. For gridded data the *pp*-function reduces to the tensor product of two one-dimensional *pp*-functions and the calculation of the coefficients becomes particularly efficient. Thus there is a distinct advantage to be had using gridded data ([5], p. 332, [8], p. 218).

To take advantage of the numerical efficiency of spline interpolation of gridded data and at the same time make more efficient use of a limited number of samples we will consider again the foveal sampling approach described in Section II. Let the image plane be decomposed into rectangular subareas. Each area is sampled using gridded data, but the sampling intervals d_x and d_y in Eq. (2.6) can be chosen differently in different subareas. Over the subareas where high resolution will be required we make d_x and d_y small, whereas over regions where lower resolution is adequate we make d_x and d_y proportionally larger. Since we intend using spline interpolation, the image need not be properly bandlimited within each subarea.

An estimate of the image is recovered from the samples in each subarea by the use of surface spline functions given by

$$B_{ij}(x,y) = B_{i,k}(x)B_{j,k}(y)\,, \tag{5.1}$$

where $B_{i,k}(x)$ and $B_{j,k}(y)$ are defined in Eq. (3.5) ([8], Eq. (3.30)). These splines are superimposed to yield a *pp*-function interpolating the data as given by

$$I(x,y) = \sum_{i,j} \alpha_{ij} B_{ij}(x,y) \ , \tag{5.2}$$

in which the α_{ij} coefficients are to be computed in a manner very similar to computation in the one-dimensional case of the last section. Computer programs exist for two-dimensional bicubic interpolation ([8], p. 221). The computation of the α_{ij} coefficients can be carried out more efficiently than by straightforward extension of the one-dimensional method to two dimensions ([5], p. 343). Some simple experiments with cubic spline interpolation of image data were done by Hou and Andrews [9,10], but their work was limited to equispaced data taken at the knots. Much more experimental work should be done to develop and extend this as a useful technique.

VI. CONCLUSIONS

The conclusions reached in this study can be stated very simply. The best method of sampling an image is by the use of Nyquist samples provided the image is properly bandlimited and gridded data are used. If the number of samples that are allowed is very limited, then the best method is to divide the image into subareas of gridded data. If the image can be properly bandlimited within each subarea, then again Nyquist sampling is best, but if the image cannot be properly bandlimited, then spline interpolation is better.

These conclusions do not hold for some unusual situations like, for example, the extreme case of absolute minimum samples and unlimited computer capability where the more general use of scattered data and general spline interpolation is probably nearer to optimum.

VII. REFERENCES

1. L. De Branges, *Hilbert Spaces of Entire Functions* (Prentice-Hall, Englewood-Cliffs, N.J., 1968).
2. W. K. Pratt, *Digital Image Processing* (John Wiley, New York, 1978).
3. F. O. Huck, N. Halyo, and S. K. Park, "Aliasing and Blurring in 2-D Sampled Imagery," Appl. Opt. **19**, 2174-2181 (1980).
4. I. J. Schoenberg, "Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions," Quant. Appl. Math. 4, 45-99, 112-141 (1946).
5. C. de Boor, *A Practical Guide to Splines* (Springer-Verlag, New York, 1978).
6. H. B. Curry and I. J. Schoenberg, "On Polya Frequency Functions IV: The Fundamental Spline Functions and Their Limits," J. d'Analyse Math. **17**, 71-107 (1966).
7. C. F. Gerald, *Applied Numerical Analysis,* 2nd ed. (Addison-Wesley, Reading, Mass., 1978).

8. L. L. Schumaker, "Fitting Surfaces to Scattered Data," in *Approximation Theory II*, G. G. Lorentz, C. K. Chui, and L. L. Schumaker, eds. (Academic, New York, 1976), pp. 203-268.

9. H. S. Hou and H. C. Andrews, "Least Squares Image Restoration Using Spline Basis Functions," IEEE Trans. C **26**, 856-873 (1977).

10. H. S. Hou and H. C. Andrews, "Cubic Splines for Image Interpolation and Digital Filtering," IEEE Trans. ASSP **26**, 508-517 (1978).

Appendix A

THE SAMPLING THEOREM

Consider a continuous function $f(x)$ defined for every x within the domain $-\infty \leqslant x \leqslant \infty$. We can represent the sampling of this function by multiplying $f(x)$ by a Dirac Comb, i.e.,

$$\overline{f}(x) = \sum_{n=-\infty}^{\infty} f(x)\delta(x - nd) \tag{A.1}$$

where d is the distance between the samples, and from Eq. (A.1) we see that

$$\begin{aligned} \overline{f}(x) &\propto f(x) \text{ if } x \text{ is an integer multiple of } d, \\ &= 0 \text{ otherwise,} \end{aligned} \tag{A.2}$$

is a collection of samples. This operation is illustrated by Fig. A.1.

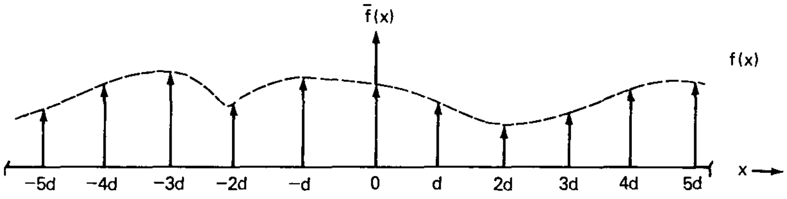

Fig. A.1 — Sampled data $\overline{f}(x)$

To transmit the sampled function $\overline{f}(x)$ we need only send the values of $\overline{f}(x)$ for $x = nd$ as a time-ordered sequence of numbers. At the receiving end we want to reconstruct the function $f(x)$ from these numbers as well as we can. Thus we require the inverse of Eq. (A.1) which gives $f(x)$ as a function of $\overline{f}(x)$.

To invert Eq. (A.1) we first take its Fourier transform, i.e.,

$$\begin{aligned} \overline{F}(\nu) &= \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(x)\delta(x - nd)\, e^{2\pi i\nu x} dx \\ &= \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} f(x)\delta(x - nd)\, e^{2\pi i\nu x} dx \\ &= F(\nu) \circledast \sum_{n=-\infty}^{\infty} e^{2\pi i\nu nd}\,, \end{aligned} \tag{A.3}$$

where we interchanged the orders of integration and summation, then used convolution theorem to remove $f(x)$ from the kernel of the integral and finally carried out the integration using the sifting property of the Dirac delta function. Consider the summation in Eq. (A.3) as a function of ν. For $\nu = m/d$ (m = any integer) we have

$$\sum_{n=-\infty}^{\infty} e^{2\pi imn} \longrightarrow \infty\,. \tag{A.4}$$

However if $\nu \neq m/d$, we have instead

$$\sum_{n=-\infty}^{\infty} e^{2\pi i\nu nd} \longrightarrow 0\,, \tag{A.5}$$

since $e^{2\pi ix}$ averages to zero. Thus, we have from Eq. (A.3)

$$\overline{F}(\nu) = F(\nu)\Theta \sum_{m=-\infty}^{\infty} \delta(\nu - m/d)\,, \tag{A.6}$$

which represents the frequency spectrum $F(\nu)$ of the original function $f(x)$ convolved with another Dirac Comb with a spacing inversely proportional to d. This is shown in Fig. A.2. If the original function was bandlimited before sampling such that $F(\nu) = 0$ outside of the domain $-1/2d \leqslant \nu \leqslant 1/2d$, then the various orders in this figure do not overlap which is the case as shown. Then to remove the effect of sampling we need only select the central order by multiplying $\overline{F}(\nu)$ by a Rect function in the manner

$$\begin{aligned} F(\nu) &= \overline{F}(\nu)\ \mathrm{Rect}\ (\nu d) \\ &= \overline{F}(\nu)\ \text{ if } |\nu| \leqslant \frac{1}{2d}\,, \\ &= 0 \text{ otherwise}\,. \end{aligned} \tag{A.7}$$

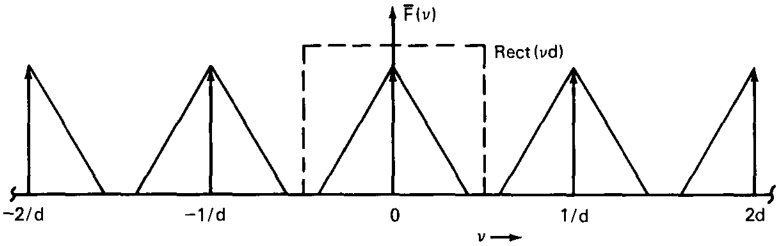

Fig. A.2 — Spectrum of sampled data $\overline{F}(\nu)$

Taking the Fourier transform of Eq. (A.7) and substituting from Eq. (A.1) we have

$$
\begin{aligned}
f(x) &= \overline{f}\,(x) \circledast \int_{-1/2d}^{1/2d} e^{-2\pi i \nu x} dx \\
&= \overline{f}\,(x) \circledast \frac{\sin\left(\dfrac{\pi x}{a}\right)}{\pi x} \qquad \text{(A.8)} \\
&= \sum_{n=-\infty}^{\infty} f(nd) \frac{\sin\left[\dfrac{\pi}{d}(x - nd)\right]}{(x - nd)} \; .
\end{aligned}
$$

Equation (A.8) shows that we can reconstruct the function $f(x)$ for *every* x from just the sampled values at $x = nd$ by convolving the samples with a sinc function spline. This is the usual sampling theorem. If the original function $f(x)$ was *not* bandlimited as assumed in Fig. A.2 so that the orders overlap as shown in Fig. A.3, then the spectrum of $f(x)$ is not simply repeated as before but high frequencies from one order are confused as lower frequencies by the next order. If we attempt to use the interpolation formula in Eq. (A.8), we obtain the function $f'(x)$ with the spectrum shown in Fig. A.4. In the region of overlap the frequencies are given by

$$\overline{F}'(\nu) = F(\nu) + F(\nu - (1/d)) \; . \qquad \text{(A.9)}$$

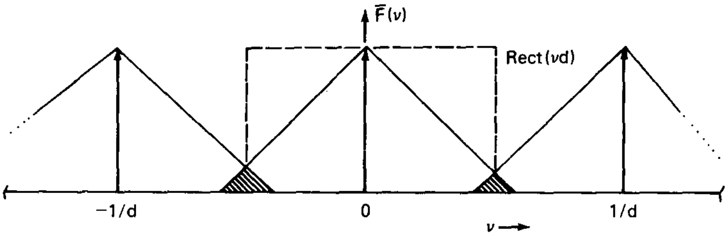

Fig. A.3 — Spectrum of sampled data $\overline{F}(\nu)$ showing aliasing

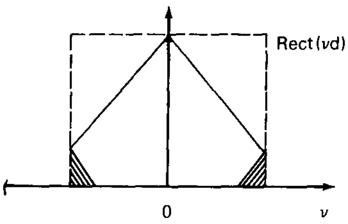

Fig. A.4 — Spectrum of the interpolated spectrum showing aliasing

The contribution from the second term in Eq. (A.9) represents beats between the sampling raster and the frequencies in $f(x)$. This appears in the interpolated function $f'(x)$ as errors termed aliasing errors. The phenomena is called aliasing. The requirement that $F(\nu) = 0$ unless $-1/d < \nu < 1/d$ which avoids aliasing is equivalent to the well-known sampling criteria that $f(x)$ must be sampled twice per period of the highest frequency component in $f(x)$. In two dimensions the sampling theorem for a rectangular array of equispaced samples can be generalized to give

$$f(x,y) = \sum_{n,m=-\infty}^{\infty} f(nd,md) \frac{\sin\left[\frac{\pi}{d_x}(x - nd_x)\right]}{(x - nd_x)} \frac{\sin\left[\frac{\pi}{d_y}(y - md_y)\right]}{(y - md_y)} . \qquad \text{(A.10)}$$

A similar procedure has been worked out for uniform sampling in circular coordinates which is rather more complicated [A1,A2].

REFERENCES

A1. H. Stark, "Sampling Theorems in Polar Coordinates," J. Opt. Soc. Am. **69**, 1519-1525 (1979).

A2. H. Stark and C. S. Sarna, "Image Reconstruction Using Polar Sampling Theorems," Appl. Opt. **18**, 2086-2088 (1979).

Appendix B

B-SPLINES REPRESENTED AS A DIVIDED DIFFERENCE

There is another representation for the *B*-splines which is often more useful than Eq. (3.1). To obtain this representation consider any n times differentiable function $h(x)$ and consider the Taylor's series representation

$$h(x) = \sum_{i=0}^{n-1} \frac{(x-a)^i}{i} h^{(i)}(a) + \int_a^x \frac{(x-t)^{n-1}}{(n-1)!} h^{(n)}(t)\, dt \,. \tag{B.1}$$

We consider a sequence of points on x (sometimes called a knot sequence) given by $x_1, x_2, \ldots x_n$ in which x_i increases monotonically with i and with a constant interval $\Delta x = x_{i+1} - x_i$. We define the first divided difference of $h(x)$ at x_i by

$$\left[x_i, x_{i+1}\right] h(\cdot) := \frac{h(x_i) - h(x_{i+1})}{x_i - x_{i+1}} \tag{B.2}$$

and similarly the kth divided difference is given by ([5], p. 8, property VIII)

$$\left[x_i, x_{i+1}, \ldots, x_{i+k}\right] h(\cdot) := \frac{\left[x_i, \ldots, x_{r-1}, x_{r+1}, \ldots, x_{i+k}\right] h(\cdot) - \left[x_i, \ldots, x_{s-1}, x_{s+1}, \ldots x_{i+k}\right] h(\cdot)}{x_s - x_r} \tag{B.3}$$

where x_r and x_s are any two different points. Thus we see from Eq. (B.3) that a kth order divided difference of $h(x)$ can be built up by taking k first order divided differences of $h(x)$. The divided difference is clearly very closely connected to the derivative of the function ([5], p. 8, property VII)). If we take the nth order divided difference of Eq. (B.1), we find that

$$\left[x_i, \ldots, x_{i+n}\right] h(\cdot) = \left[x_i, \ldots, x_{i+n}\right] \int_a^{(\cdot)} \frac{(\cdot - t)^{n-1}}{(n-1)!} h^{(n)}(t)\, dt \tag{B.4}$$

where we have used the property ([5], p. 6, property V)

$$\left[x_i, \ldots, x_{i+n}\right] (x-a)^m = 0 \text{ if } m < n \,, \tag{B.5}$$

which can be proven by direct application of Eq. (B.3). If we define

$$(x - t)_+^{n-1} = (x - t)^{n-1}, \text{ if } (x - t) \geqslant 0 \,, \tag{B.6}$$

$$= 0 \text{ otherwise,}$$

then we have (for $b > x$)

$$\int_a^x \frac{(x - t)^{n-1}}{(n - 1)!} h^{(n)}(t)dt = \int_a^b \frac{(x - t)_+^{n-1}}{(n - 1)!} h^{(n)}(t)dt \tag{B.7}$$

and by substitution into Eq. (B.4) we find that

$$\Big[x_i, \ldots, x_{i+n}\Big] h(\cdot) = \int_a^b \frac{\Big[x_i, \ldots, x_{i+n}\Big] (\cdot - t)_+^{n-1}}{(n - 1)!} h^{(n)}(t)dt \,. \tag{B.8}$$

Now if we let

$$h(x) = e^{i\xi x}$$

$$a = -\infty$$

$$b = \infty \,, \tag{B.9}$$

Eq. (B.8) becomes

$$\Big[x_i, \ldots, x_{i+n}\Big] e^{i\xi x} = (i\xi)^n \int_{-\infty}^{\infty} \frac{\Big[x_i, \ldots, x_{i+n}\Big] (\cdot - t)_+^{n-1}}{(n - 1)!} e^{i\xi t}\, dt \,. \tag{B.10}$$

The RHS of Eq. (B.10) is just the Fourier transform of the nth divided difference of $(x - t)_+^{n-1}$ and the LHS can be evaluated using (B.3) to give

$$\Big[x_i, \ldots, x_{i+n}\Big] e^{i\xi x} = \frac{(i\xi)^n}{n!} e^{i\xi \bar{x}} \left(\frac{\sin (\Delta x \xi/2)}{\Delta x \xi/2} \right)^n \,, \tag{B.11}$$

when $\Delta x = x_{i+1} - x_i$, and where $\bar{x} = (x_{i+n} + x_i)/2$ is the middle point of the sampling domain.

By substitution from (B.11) into (B.10) we have

$$\left[\frac{\sin (\Delta x \xi/2)}{\Delta x \xi/2} \right]^n = n \int_{-\infty}^{\infty} \Big[x_i, \ldots, x_{i+n}\Big] (\cdot - t)_+^{n-1} e^{i\xi(t - \bar{x})}\, dt \,, \tag{B.12}$$

which by Fourier inversion becomes

$$n\left[x_i, \dots, x_{i+n}\right](\cdot - t)_+^{n-1} = \frac{1}{2\pi}\int_{-\infty}^{\infty}\left[\frac{\sin(\xi/2)}{\xi/2}\right]^n e^{-i\xi t}d\xi \tag{B.13}$$

where we have rescaled x so that $\Delta x = 1$ and $\bar{x} = 0$. By comparison of Eq. (B.13) with Eq. (3.1) we have

$$M_k(t) = k\left[x_i, \dots, x_{i+k}\right](\cdot - t)_+^{k-1}\,, \tag{B.14}$$

the kth spline is the kth divided difference of the function $k(x - t)_+^{k-1}$. This is the relation that is almost always used in the mathematical literature to define the B-splines.

Although the original definition of B-splines as given in Eq. (3.1) holds only for a sequence of equally spaced knots the definition in Eq. (B.14) can be generalized to an arbitrary knot sequence ([5], p. 108), i.e.,

$$B_{i,k}(t) = \left(\tau_{i+k} - \tau_i\right)\left[\tau_i, \dots, \tau_{i+k}\right](\cdot - t)_+^{k-1}\,, \tag{B.15}$$

where the knot sequence $\tau_i, \tau_{i+1}, \dots, \tau_{i+k}$ are arbitrary points on a line (in any order, with any spacing, and possibly with repeated values).

By applying Leibniz' formula for the kth divided difference of a product to the particular product

$$(t - x)_+^{k-1} = (t - x)(t - x)_+^{k-2} \tag{B.16}$$

we can show ([5], p. 130) that the generalized B-splines obey the recurrence relation

$$B_{i,k}(t) = \frac{t - t_i}{t_{i+k-1} - t_i}B_{i,k-1}(x) + \frac{t_{i+k} - t}{t_{i+k} - t_{i+1}}B_{i+1,k-1}(t)\,. \tag{B.17}$$

From Eq. (B.17) and the evident fact (from Eq. (B.15)) that

$$\begin{aligned} B_{i,1}(t) &= 1 \text{ if } t_i \leqslant t \leqslant t_{i+1}\,, \\ &= 0 \text{ otherwise,} \end{aligned} \tag{B.18}$$

we have a simple way of generating the B-splines in a computer. Because of the complexities of dealing with divided differences we use Eqs. (B.17) and (B.18) to define generalized B-splines in this report.